Application of Musical Computing to Creating a dynamic re-configurable Multi-layered Chamber Orchestra Composition

Alexis Kirke (composer, researcher), University of Plymouth, Interdisciplinary Centre for Computer Music Research, Room 003, Hepworth Building, Drake Circus, UK. Email: alexis.kirke@plymouth.ac.uk, ORCID 0000-0001-8783-6182

Abstract

With increasing virtualization, and recognition that today's virtual computers are faster than hardware computers of 10 years ago, modes of computation are limited only by imagination. Pulsed Melodic Affective Processing (PMAP) is an Unconventional Computation protocol which makes affective computation more human-friendly by making it audible. Data sounds like the emotion it carries. PMAP has been demonstrated in non-musical applications, e.g. quantum computer entanglement and stock market trading. This paper presents a musical application of, and demonstration of, PMAP: a dynamic reconfigurable score for acoustic orchestral performance, in which the orchestra acts as a PMAP half-adder to add two numbers.

Orchestras and Non-musical Processes

This article presents a composition in which an orchestra carries out a non-musical process, specifically a form of computation. Most orchestral projects involving non-musical processes are musification (i.e. using more traditional elements of sound such as melody, harmony, and rhythm). The composition Orchestral Processing Unit discussed here is neither. Orchestral Processing Unit is an instantiation of a computation. However, because most orchestral work related to non-musical processes is musification, examples of past work in that area are briefly highlighted.

Cullen and Coyle [1] used synthesized orchestration to sonify a database of employee related data in a database. Hinterberger [2] examines the sonification of EEG ("brainwave") readings with orchestra. There has been an orchestral-sound project [3] that sonifies data from supernovae. Another sonification of physics data is the LHChamber Music project [4]. Different instruments played data from different experiments. Eduardo Miranda [5] wrote a piece for symphonic orchestra and choir, as part of which he sonified the output of a simulated biological neural network. The Heart Chamber Orchestra [6] generated a live score using the heart rates of 12 musicians. It is put on computer screens in front of each musician.

The process used to generate the acoustic orchestral performance in this paper is computation. What is unusual about this is that the computation itself in a musical substrate. The motivation and implementation of this substrate is now explained.

Musical Processes with Non-musical Function

Virtual Machines have existed for decades, such as the Java Virtual Machine [7], and those which allow users to run a non-native OS [8]. Server virtualization is common. It sacrifices processing power in the virtualization process [9], but this is outweighed by the advantages.

The virtual machine referred to in this article does not reduce computation time. It is more in the philosophy of the graphical user interface (GUI) interaction paradigm, also known as Window, Icon, Mouse and Pointer (WIMP) interfaces. WIMP did not increase processing power when added. In fact they reduced it [10] because of the requirements for bitmapped screens and windows. However many tasks now would be unfeasibly slow for us without WIMP. Changing the mode of human-computer interaction opens up opportunities for increasing the usefulness of computers, even though it uses up processing power.

Given the growth in virtual computing, unconventional computing has the opportunity to greatly expand its possible modes, limiting computation only by imagination; hence the field of Unconventional Virtual Computation (UVC) [11]. Previous work using simulations to run unconventional computation was to simulate a hardware or wetware system, not to the make fundamental simulation the substrate (e.g., L. Spector et al. "Finding a better-than-classical quantum AND/OR algorithm using genetic programming," in Proceedings of the 1999 Congress on Evolutionary Computation (Washington DC: IEEE, 1999) pp 2239--2246.)

One mode of UVC that could provide benefits is Human-Computer Interaction by Replacement (HCI By replacement, or HBR). HBR [13] is an approach to unconventional virtual computing that combines computation with HCI, a complementary approach in which computational efficiency and power are more balanced with understandability to humans. Rather than ones and zeros in a simulated circuit, have the user-interface object itself; e.g. if you want data to be audible, replace the computation substrate by melodies. Some forms of HBR may not be implementable in hardware in the foreseeable future, but current hardware speeds could be matched by future virtual HBR machines.

The focus here will be on forms of HBR in affective computation or in computation that has an affective interpretation. It has been shown that affective states (emotions) play a vital role in human cognitive processing and expression [14]. As a result, affective state processing has been incorporated into robotics and multi-agent systems. Representing and computing with affective states is an active area of research.

The dimensional approach to specifying emotional state is one common approach. In many emotional music systems [15] two dimensions are used: valence and arousal. In that model, emotions are plotted on a graph with the first dimension being how positive or negative the emotion is (Valence), and the second dimension being how intense the physical arousal of the emotion is (Arousal). For example “Happy” is high valence, high arousal affective state, and “Stressed” is low valence, high arousal state.

To a degree these can be represented musically. There have been a number of questionnaire studies done which support the argument that music communicates emotions and previous research [16] has suggested that a main indicator of valence is musical key mode. A major key mode implies higher valence, minor key mode implies lower valence. For example the overture to Mozart’s The Marriage of Figaro is in a major key whereas the melancholic first movement of Beethoven’s Piano Sonata No. 14 (Moonlight) is in a minor key. That research also highlights that tempo is a prime indicator of arousal, with high tempo indicating higher arousal, and low tempo - low arousal. For example: compare Mozart’s fast overture above with Debussy’s major key but low tempo opening to “Girl with the Flaxen Hair”. The Debussy piano-piece opening has a relaxed feel, that is a low arousal despite a high valence. This is a rather coarse analysis, and further details can be found in, for example [Kirke thesis paper]

PMAP Representation

In the HBR protocol Pulsed Melodic Affective Processing (PMAP) [17] the data stream representing affective state is a stream of pulses transmitted at a variable rate (c.f. the variable rate of pulses in biological neural networks in the brain). The pulse rates encode information (neuroscientists often use audio probes to listen to neural spiking). In PMAP this pulse rate specifically encodes a representation of the arousal of an affective state. A higher pulse rate is essentially a series of events at a high tempo (hence high arousal); whereas a lower pulse rate is a series of events at a low tempo (hence low arousal).

Additionally, the PMAP pulses can have variable heights with 10 possible levels. For example 10 different voltage levels for a low level stream, or 10 different integer values for a stream embedded in some sort of data structure. The purpose of pulse height is to represent the valence of an affective state, using pitch and keymode. There is not space to explain this here and it is not used in the example covered in this article; for more details and examples that use keymode *and* tempo, see [18].

PMAP provides a method for the processing of artificial emotions. PMAP data tempo can be generated directly from rhythmic data and turn directly into rhythmic data or sound. Thus rhythms such as heart rates, key-press speeds, or time-sliced photon-arrival counts can be directly turned into PMAP data; and PMAP data can be directly turned into music with minimal transformation. PMAP has been applied and tested in a number of simulations. More details are available in previous UVC and PMAP publications mentioned:

a. A security team multi-robot system [17]

b. A musical neural network to detect textual emotion [17]

c. A stock market algorithmic trading and analysis approach [19]

d. A system that keeps a photonic quantum computer in a state of maximum entanglement [11].

Due to lack of space, only (a) will be briefly described. The security robot team simulation involved robots with two levels of intelligence: a higher level more advanced cognitive function and a lower level basic affective functionality. The lower level functionality could take over if the higher level ceased to work. A new type of logic gate was designed to use to build the lower level: musical logic gates. PMAP equivalents of AND, OR and NOT were defined, inspired by Fuzzy Logic.

The PMAP versions of these are respectively: MAND, MOR and MNOT (pronounced “emm-not”), MAND, and MOR. So for a given stream, a PMAP segment of data can be summarized as $m_i = [k_i, t_i]$ with key-value k_i and tempo-value t_i. The definitions of the musical gates are (for two streams m_1 and m_2):

$$\mathrm{MNOT}(m) = [-k, 1-t] \quad (1)$$

$$m_1 \text{ MAND } m_2 = [\min(k_1,k_2), \min(t_1,t_2)] \quad (2)$$

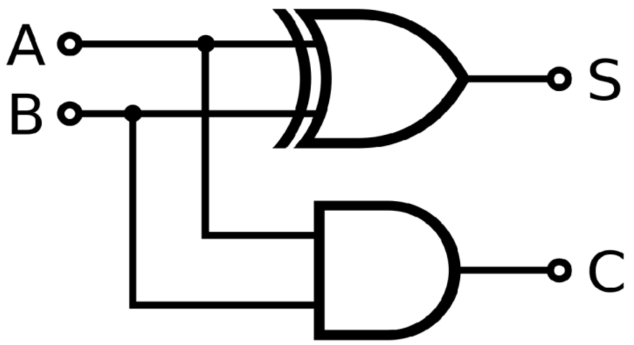
$$m_1 \text{ MOR } m_2 = [\max(k_1,k_2), \max(t_1,t_2)] \quad (3)$$

It was shown that using a circuit of such gates, PMAP could provide basic fuzzy search and destroy functionality for an affective robot team. It was also found that the state of a three robot team was human audible by tapping in to parts of the PMAP processing stream, and listening to keymode and tempo.

PMAP Half-Adder

The main focus of this article is the application of PMAP to a dynamic re-configurable orchestral performance. “Orchestral Processing Unit” (OPU) is a chamber orchestra performance using the tempo encoding of PMAP. Because of an orchestra’s size and slow reaction, and the need for a performance to have enjoyable coherent development, it was decided that an orchestral performance using PMAP would have to focus on the PMAP process, rather than the PMAP process controlling something else. Furthermore it was desired to make the process a form of computation that was understandable to the audience in a more intuitive way. So using an orchestra as an affective computation system would be too complex for the audience to understand and therefore experience. Thus a more traditional form of computer circuit was implemented in orchestral PMAP: a two-bit addition system. Such a system, when looped through twice, allows the addition of two numbers between 0 and 3.

The orchestral PMAP adder is implemented using two passes through a half-adder. A half-adder is implemented in traditional computation as seen in Figure 1. S is the sum of binary numbers A and B. If the resulting sum is greater than 1 bit in size, it is set to 0 and the carry flag C is set to 1. A half-adder allows two numbers between 0 and 1 to be added as shown in Figure 1. This is why two passes are needed through the PMAP half-adder: to allow numbers between 0 and 2 to be added.



Input A (A)	Input B (B)	Sum (S)	Carry (C)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Fig. 1. Circuit and truth table for a half-adder (Circuit attribution: inductiveload [Public domain], from Wikimedia Commons. Truth Table (c) 2019 Alexis Kirke)

The most musically controllable way of making a half-adder in PMAP is to focus on two tempo values, and remove the pitch dimension. It is problematic to use the pitch dimension in a musical composition as having multiple lines playing in the orchestra in which one is major and one is minor will usually lead to unpleasant dissonance.

For OPU a logic level is represented by a monophonic musical phrase. If the phrase is played with its original note lengths it is considered to be logical 1, if played at half its original note length, it is considered as logic 0. Then the AND gate in Figure 1 can be replaced by a

MAND gate. The XOR gate is replaced as followed. XOR is constructed in Boolean logic using AND, OR and NOT gates:

$$Output = (A\ AND\ NOT(B))\ OR\ (NOT(A)\ AND\ B) \tag{4}$$

Because the MAND is functionally a two dimensional Fuzzy AND gate, and because the tempo in OPU has only two states: a minimum and a maximum, and because in the extreme a Fuzzy AND is identical to an AND gate: the AND gate can be replaced by a MAND gate. Similarly for the OR and NOT gates. Thus we have:

$$Output = (A\ MAND\ MNOT(B))\ MOR\ (MNOT(A)\ MAND\ B) \tag{5}$$

Equation (2) is then implemented in the orchestra as shown in Figure 2.

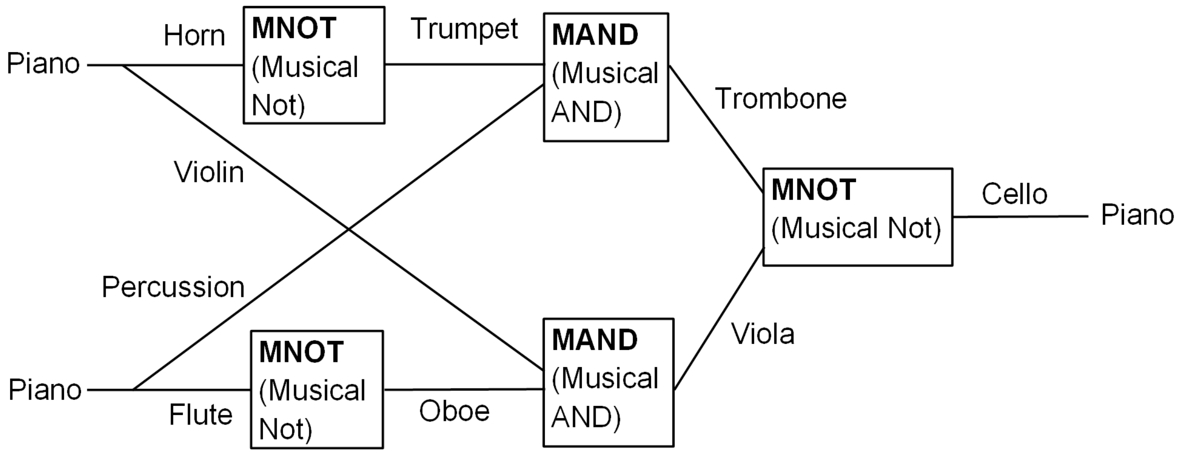


Fig. 2. PMAP half-adder, looped through twice ((c) 2019 Alexis Kirke)

Procedure in Performance

The processing is implemented as follows. Melodies – call them M_A and M_B – represent inputs A and B in Figure 1. For M_A a logical 1-value is represented by M_A played with its defined note lengths. A 0-value is represented by M_B played with its note lengths at half the note length of M_A – i.e. the melody is played at twice the speed, and takes half as long to play. Figure 3 shows the two possible states for one of the clarinets during a particular part of the calculation / performance. The upper line is logic 1 and the lower line (with twice the note lengths) is logic 0 (only first half shown).

Fig. 3. Examples of logic 1 and logic 0 from part of the clarinet score ((c) 2019 Alexis Kirke)

Each group of instruments in Figure 2 has a paper score showing, at every point in the performance, the two possible phrases they could be playing: the maximum tempo phrase, and same phrase played at half the tempo. The conductor indicates to players when they should play and whether they should play the top line or the bottom line.

Note - due to the size of the orchestra, and the time and style constraints - the conductor performs the carry calculations and storage – these are not included in Figure 2. As will be seen later, the conductor is given a rule to say that if both the trombone and viola represent a 1-bit, then the conductor should make a note of this. The existence or non-existence of these notes will effect which musical lines they indicate to the players later in the calculations.

To see how one of the two half-adder calculations are done, suppose the conductor wants to input a 1-value to the top input in the circuit in Figure 2. The piano is the input, output and storage register of the adding process (it does not perform any processing). So the conductor will indicate the piano should start playing M_A. Consider the top line of processing in Figure 2, which can be written as:

$$Horn = Piano \qquad (6)$$

$$Trumpet = NOT(Horn) \qquad (7)$$

The conductor next indicates to the horn that they should play what the piano (the input register) is playing from their score. The horn plays M_A as well. The next stage is the trumpet playing the MNOT of M_A. The definition of MNOT is that the maximum tempo becomes the minimum tempo, and vice versa (equation 1). So the conductor will indicate to the trumpet to start to play M_A at double note length (half the speed). The next musical gate along this line of Figure 2 is the MAND. Based on equation (2), the MAND will output the minimum tempo for all inputs, unless both of its inputs are maximum tempo. This MAND gate has inputs from the percussion and the trumpet. Given the trumpet is playing minimum tempo, the MAND output – the Trombone – will be told by the conductor play the phrase at a minimum tempo. The whole trombone score is shown in Figure 4.

Fig. 4. Single Instrument Score Example – Tenor Trombone ((c) 2019 Alexis Kirke)

The conductor was given a "verbal score" reproduced in Figure 5 to perform the calculation. The first column – the segment column - refers to markings on the conductor's full score. This score is too large to include clearly, but an excerpt is shown in Figure 6. Segments S1 to S29 perform two XOR calculations. S31 is actually a special case for the conductor instruction set, as shown in Figure 7. This involves combining carries to create the final sum. These carries are noted down by the conductor during the playing of Segments S4 and S19. The pianos are also a special case. As the calculation / performance, continues, the pianos act as registers – storing intermediate calculation results. Thus the piano can store two bits of data: in the left and the right hand.

Looking at Figure 5, the four bits are input at segments S1, S3, S16 and S18. S1 and S3 are the least and most significant bit for the first input; S16 and S18 are the least and most significant bit for the second input. After the conductor guided through to S31, the piano plays the three output bits.

Performing a Calculation in Concert

The performance took place at the Peninsula Arts Contemporary Music Festival, performed by the Ten Tors Orchestra and conducted by Simon Ible. A recording of the PMAP movement is available online [20], starting at 6m15s. The two numbers added in the performance were 3 and 3. These were sent to the conductor around 48 hours before the performance (he had had the re-configurable score for a few weeks before this). The binary values were 11 and 11, so what was actually sent to the conductor were the instructions to get the piano to play the top line of the score at each of its four input points. The result of the sum meant that two carries of 1 were stored by the conductor, and this led to him instructing the pianist to play certain chords at the end whose constituent lines represented binary 110 – which is 6 in decimal, the result of 3+3.

The orchestra succeeded in calculating accurately. The author did not attend the rehearsals, providing only scores and instructions. The accurate performance indicated the musicians' ability to interpret a computation with the conductor. The fact the conductor asked for the calculation inputs to be provided 48 hours before the performance indicates that he wanted to limit the degree to which the musicians had to make live choices in responding to him during live performance. However he was never given the "hard" score – i.e. the actual fixed musical score for a 3+3 calculation, but only the protocol and the inputs.

Is such a performance is enjoyable to an audience, or just a curiosity? Apart from positive informal audience feedback there was a positive review by a music critic [21]:

"Binary code is reinterpreted as musical melodies, and the Ten Tors orchestra pulsate in a powerfully pretty and pointillistic Reichian performance piece…Orchestral Processing Unit holds its own as an intensely beautiful experience."

This is one person's opinion but supports the anecdotal feedback from the general audience.

Segment	**Instrument Beginning**	**Top or Bottom?**
S1	Piano	Composer will tell you top X or bottom X stave
S2	Violin	Same stave as Piano X in S1
S3	Piano	Composer will tell you top X or bottom X stave
S4	Flute (If violin and flute are both top, conductor make a note somewhere say **"C1"**)	Same stave as Piano X in S3
S5	[Piano STOP] Oboe	If Flute playing top, then play bottom. If flute playing bottom, then play top.
S6	Viola	Top if violin and oboe is top, bottom otherwise
S7	Piano	Composer will tell you top X or bottom X stave
S8	Horn	Same stave as Piano X in S7
S9	Piano	Composer will tell you top X or bottom X stave
S10	Timpani	Same stave as Piano X in S9
S11	[Piano STOP] Trumpet	If horn playing top, then play bottom. If horn playing bottom, then play top.
S12	Trombones	Top if Timpani and Trumpet are top, bottom otherwise
S13	Cello	Top UNLESS trombone and viola are bottom
S14	Piano	Same stave as Cello in S13
S15	Stop all instruments EXCEPT Piano	
S16	Piano Y	Pianist keeps playing X stave it was playing in S15 all the way through to S31 inclusive. Instructions below are for *additional* lines. Composer will tell you top Y or bottom Y stave
S17	Violin	Same stave as Piano Y in S16

S18	Piano Y	Composer will tell you top B or bottom B stave
S19	Flute / Bass / Clarinet 1 & 2 / Oboe (If violin and flute are both top, conductor make a note somewhere say **"C2"**)	Same stave as Piano Y in S18
S20	Bassoon	If Flute playing top, then play bottom. If flute playing bottom, then play top.
S21	Viola	Top if violin and bassoon are top, bottom otherwise.
S22	Piano Y	Composer will tell you top Y or bottom Y stave
S23	Horn	Same stave as Piano Y in S22
S24	Piano Y	Composer will tell you top Y or bottom Y stave
S25	Timpani	Same stave as Piano Y in S24
S26	[Piano STOP, except for X stave theme] Trumpet	If horn playing top, then play bottom. If horn playing bottom, then play top.
S27	Trombone / Bass Trombone	Top if Timpani and Trumpet are Top, Bottom otherwise
S28	Cello	Top UNLESS trombone and viola are bottom, in which case bottom.
S29	Piano Y	Same stave as Cello in S28
S30	Stop all instruments EXCEPT Piano	
S31	Piano X, Y, Z	LEFT HAND: Continue Piano X one octave down. RIGHT HAND (See table below or): If C2 marked then following applies to Piano Z, otherwise it applies to Piano Y: If Piano was playing Piano Y top in S30 and C1 is marked, OR if Piano was playing Piano Y bottom in S30 and C1 is not marked, then play bottom; else play top.
S32	Stop all instruments	

Fig. 5. Conductor Verbal Score ((c) 2019 Alexis Kirke)

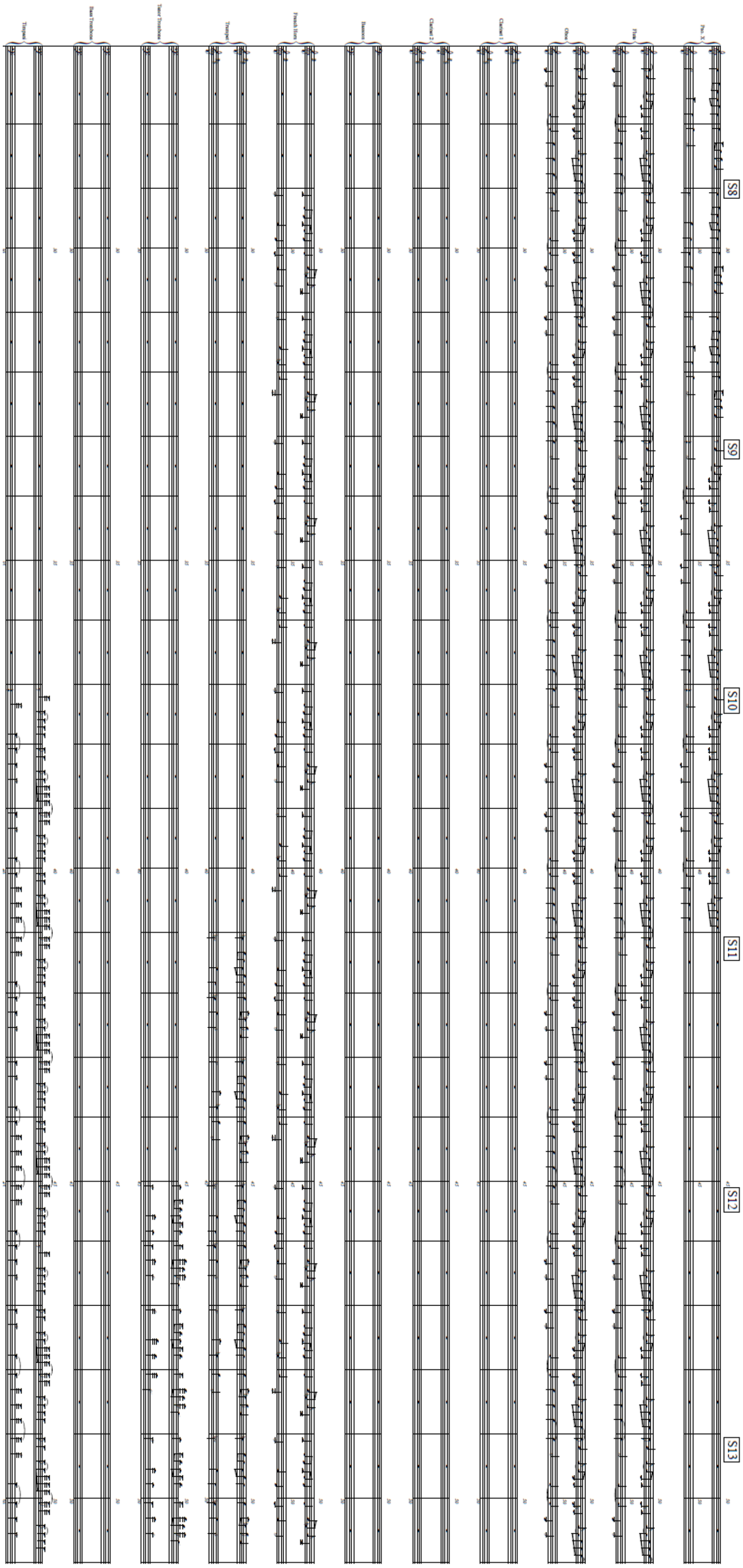


Fig. 6. Conductor Score Excerpt ((c) 2019 Alexis Kirke)

Conductor selection table for S31

This table is an alternative to using the conductor instructions in S31. It may be easier to read and enact. It has the same effect as the instructions in S31 but explains them in a different way.

Continue Piano X in S30 one octave down, and:

If C2	and C1	and Piano Y in S30 playing	then Play
Not marked	Not marked	Bottom	Bottom on Piano Y
Not marked	Not marked	Top	Top on Piano Y
Not marked	Marked	Bottom	Top on Piano Y
Not marked	Marked	Top	Bottom on Piano Y
Marked	*Not marked*	*Bottom*	*Bottom on Piano Z*
Marked	*Not marked*	*Top*	*Top on Piano Z*
Marked	*Marked*	*Bottom*	*Top on Piano Z*
Marked	*Marked*	*Top*	*Bottom on Piano Z*

Fig. 7. Alternative Conductor Instructions for Final Calculation ((c) 2019 Alexis Kirke)

Conclusions

This article introduced the Unconventional Virtual Computation protocol PMAP. PMAP has been demonstrated previously to have non-musical functionalities. Orchestral Processing Unit (OPU) demonstrated OPU and examined if it could be used artistically. OPU instantiated two half-adders using tempo-based PMAP across orchestral instruments. A protocol was provided to the conductor, describing how to control the orchestra live as a PMAP circuit, and the inputs (3 and 3) were provided 48 hours before the performance. The orchestra was able to perform the calculation, whilst playing well, and the resulting performance was judged favourably by observers.

References and Notes

1. E. Coyle and C. Cullen, "Orchestration within the Sonification of basic Data Sets," in *Proceedings of the 10th Meeting of the International Conference on Auditory Display* (Sydney: ICAD, 2004).

2. T. Hinterberger, "Orchestral sonification of brain signals and its application to brain-computer interfaces and performing arts," in *Proceedings of the 2nd International Workshop on Interactive Sonification* (York: ISAD, 2007).

3. Geoff Brumfiel, "Dying stars write their own swan songs," NPR, 10 January 2014, < https://www.npr.org/sections/thetwo-way/2014/01/10/261397236/dying-stars-write-their-own-swan-songs>, accessed 1 November 2018.

4. Sophie Hetherton, "Cern scientists perform their data," CERN, 5 April 2017, < https://home.cern/about/updates/2014/10/cern-scientists-perform-their-data>, accessed 1 November 2018.

5. Eduardo Miranda, *Thinking Music* (Plymouth: Plymouth University Press, 2014) p. 92.

6. P. Votava and E. Berger "The Heart Chamber Orchestra," *eContact!* **14**, No. 2 (2012).

7. S. Freund and J. Mitchell, "A formal framework for the Java bytecode language and verifier," in *Proceedings of the 14th ACM SIGPLAN conference* (New York: ACM, 1999) pp. 147--166.

8. "Understanding Full Virtualization, Paravirtualization, and Hardware Assist", VMWare Inc, 2007, < http://www.vmware.com/files/pdf/VMware_paravirtualization.pdf >, Accessed 1 November 2018.

9. W. Guohui and T. Ng, "The Impact of Virtualization on Network Performance of Amazon EC2 Data Center," in *Proceedings of 2010 IEEE INFOCOM* (San Diego: IEEE, 2010) pp.1--9.

10. J. Garmon, "What were the original system requirements for Windows 1.0?", November 2010, < http://www.jaygarmon.net/2010/11/what-were-original-system-requirements.html>, accessed 1 November 2018.

11. A. Kirke et al. "A Hybrid Computer Case Study for Unconventional Virtual Computing," *International Journal of Unconventional Computing* **11**, No. 3-4, 205--226 (2015).

13. See Kirke et al. [11].

14. S. Banik et al. (2008) "Affection Based Multi-robot Team Work," in *Lecture Notes in Electrical Engineering* (Berlin: Springer, 2008) pp. 355--375.

15. A. Kirke and E. Miranda, "A Multi-Agent Emotional Society Whose Melodies Represent its Emergent Social Hierarchy and Are Generated by Agent Communications," *Journal of Artificial Societies and Social Simulation* **18**, No. 2 (2015).

16. P. Juslin, "From Mimesis to Catharsis," in *Music Communication* (Oxford: Oxford University Press, 2005) pp. 85--116.

17. A. Kirke and E. Miranda, "Pulsed Melodic Affective Processing," *Simulation* **90**, No. 5, 606--622 (2015).

18. See Kirke and Miranda [17]

19. A. Kirke and E. Miranda, "Application of Pulsed Melodic Affective Processing to Stock Market Algorithmic Trading and Analysis," in *Proceedings of 9th International Symposium CMMR 2012* (London: Springer, 2012).

20. A. Kirke, "Orchestral Processing Unit (premiere),", YouTube, 1 June 2017, https://www.youtube.com/watch?v=oPF0DCSXRqI, accessed 1 November 2018.

21. T. Bath, "The Musification of Biology," Drowned in Sound, 9 March 2015, <http://drownedinsound.com/in_depth/4148776-the-musification-of-biology--a-festival-of-music-and-science>, accessed 1 November 2018.

Kirke, Application of Musical Computing to Creating a dynamic re-configurable Multi-layered Chamber Orchestra Composition

Alexis Kirke is a senior research fellow at University of Plymouth in the Interdisciplinary Centre of Computer Music Research. He received his two PhDs from the Faculty of Arts (2011) and the Faculty of Technology (1997) at the University of Plymouth.